Association for Information Systems
AIS Electronic Library (AISeL)

MCIS 2007 Proceedings

Mediterranean Conference on Information Systems (MCIS)

2007

REPRESENTATION OF MULTI-STRUCTURED Documents with OWL. Applications to Philology.

Suha Kaouk
LIRIS-INSA of Lyon, kaouk.souha@gmail.com

Sylvie Calabretto
LIRIS-INSA of Lyon, sylvie.calabretto@insa-lyon.fr

REPRESENTATION OF MULTI-STRUCTURED Documents with OWL. Applications to Philology.

Kaouk, Suha, LIRIS-INSA of Lyon, 7 avenue Jean Capelle, 69621 Villeurbanne, France, kaouk.souha@gmail.com

Calabretto, Sylvie, LIRIS-INSA of Lyon, 7 avenue Jean Capelle, 69621 Villeurbanne, France, sylvie.calabretto @insa-lyon.fr

Abstract

Multi-Structured documents (denoted MSDs) are documents whose structure is composed of a set of concurrent hierarchical structures. Many distinct structures may be defined simultaneously for the same original document (logical structure, physical structure). Each hierarchy analyses the text within the document by a different point of view, which depends on different use of that text. These structures may overlap over the document contents.

XML has become the most used language for encoding electronic documents. XML documents are tree based; and since there are overlapping between different structures, the hierarchy of a tree allows encoding a document depending on one structure.

Some applications need to consider more than one hierarchy over the same text, which corresponds to different analysis for different uses of that document. If several different structures should be represented, the solution that manages several different versions for same information is not only ineffective and expensive in time and resources, but does not allow, for example, a search for information relating to two different structures for the same document.

One of the distinguished solutions that addressed this problematic, is a generic model called Multi-Structure Document Model (MSDM), which is independent of any formalism of encoding. However MSDM is encoded by formalism called MultiX that uses XML syntax. MultiX could serialize the MSDM model into XML syntax and expresses the different structures and their correspondences in a single xml file. However it still has some complexity due to its respect to XML tree model.

In this paper, we will present how to encode MSDs depending on MSDM but by means of non-tree based data model (graph based). We will use Ontology Web Language (OWL) to represent the metadata that corresponds to XML schema in MultiX. To illustrate our work, we choose, as running example, an application of philology (science dedicated to the study of text history).The example is a fragment of an old manuscript written in Occitan language.

Keywords: Multi-Structured documents, XML, MSDM, MultiX, OWL, encoding manuscripts.

1 INTRODUCTION

Multi-Structured documents are documents whose structure is composed of a set of concurrent hierarchical structures. Each hierarchy analyses the text within the document by a different point of view, which depends on different use of that text.
For example, let's take a book as a document; we could define:

- Physical structure (book structure) that deals with cover, pages, columns, lines.
- Logical structure (text structure) that deals with Chapters, Paragraphs and words.

A textual encoding (Tummarello 2007) means to specify a set of markers (or tags) which are added to the electronic representation of the text in order to define textual features , e.g., single words, lines, pages, chapters etc. These annotations make it possible for machines to perform useful tasks, e.g. advanced searching (e.g. retrieving "the average number of complete sentences in a page"). Inserting explicit markers for features in the text is often referred to 'markup', 'encoding' or 'tagging'.

1.1 Problematic

XML has become the most used language for encoding electronic documents. XML documents are tree based; and since there are overlapping between different structures, the hierarchy of a tree allows encoding a document depending on one structure (which means one analysis, one point of view).
Some applications need to consider more than one hierarchy over the same text, which corresponds to different analysis for different uses of that document. If several different logic designs should be represented, the solution that manages several different versions for same information is not only ineffective and expensive in time and resources, but does not allow, for example, a search for information relating to two different structures for the same document.
Therefore there exists now several works on concurrent markups that propose to include the different hierarchies of a document into one document (the multi-structured document) and query it. Due to the tree base of XML documents some of these works and studies, proposed extensions for Xpath (McQueen 2000), others introduced a new markup language (LMNL) (Tenison 2002).
A more revolutionist idea is to encode documents by means of graph instead of tree; whereas the graph approach would overcome easily the problem of overlapping.
Theoretically speaking, semantic web tools and languages such as RDF (Recourse Description Framework) and OWL could be an interesting solution to deal with tasks traditionally performed with XML markup tools.

1.2 Motivation example

To illustrate our work, we choose, as running example, an application of philology (science dedicated to the study of text history). The example is a fragment of an old manuscript written in Occitan language (figure1).

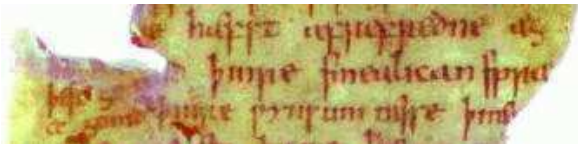

FIGURE 1. Extract of the manuscript

Following is the textual extraction of the manuscript:

…hu þu me hæfst afrefredne ægþer ge mid þinre smealican spræce ge mid þinre wynsumnesse þines…

Let's analyze the previous manuscript depending on the two hierarchies mentioned in the introduction (Physical and Lexical) Figures 2-3.

```
<?xml version="1.0" encoding="UTF-8"?>
 <Manuscript >
  <Page >
   <lines>
    <line n="1">hu u me hوfst afrefredne وg</line>
    <line n="2">er ge mid inre smealican sprو</line>
    <line n="3">ce, ge mid inre wynsumnesse ines</line>
   </lines>
  </Page >

 </ Manuscript >
```

FIGURE 3. Physical structure

```
<?xml version="1.0" encoding="UTF-8"?>
<Syntax >
 <words>
   <w>hu</w><w>u</w><w>me</w><w>hوfst</w><w>afrefredne</w><w>وger</w>
    <w>ge</w> <w>mid</w><w>inre</w><w>smealican</w><w>sprوce</w><w>ge</w>
    <w>mid</w>   <w>inre</w><w>wynsumnesse</w><w>ines</w>
 </words>

 </ Syntax >
```

FIGURE 4. Lexical structure

As we can see, each structure is independent from the other. It can be described by an XML DTD, or XML schema…
We can also notice the overlapping between the content of these two structures. For example, the word "ægþer" (Lexical structure) is cut between the first two lines (physical structure).
We are interested in taking into account simultaneously these structures in order to model and query them. In particular, we would like to be able to express queries like: *What are the words cut by an end of line?*

2 RELATED WORKS

From the problematic explained in the previous example, works and studies started. We can classify those approaches in three categories:

- To maintain each structure in a separate document and keep the hierarchy of tree-like in XML files such as MXSD (Bruno 2006). This approach is costly (redundancy of contents); in addition, in a case of querying among structures, it would be difficult to navigate from one hierarchy to another.
- To generate one document containing all structures (i.e. the multi-structured document) such as MSDM (Chatti 2006).
  The challenge imposed by these kinds of approaches is that by using available XML tools, the document should be hierarchical, and this is not the case due to the overlapping problem among different structures.
- Similar to the previous kind but by using non-XML approaches: generating new ones such as LMNL or using semantic web tool such as textual encoding based on RDF (Tummarello 2005).

3 OWL FOR ENCODING MSDs

We need a model that covers the following features:

- Depends on graph model.
- Limits redundancy of contents.
- Defines boundaries between structures, hence enables their reusability.
- Provides validation of MSDs according to a schema or constraints.
- Could be extended to new concepts (multimedia).

From the previous state of the art we found that Multi-Structure Document Model (MSDM) (Chatti 2006) is a generic model that might take into consideration all the previous points.

3.1 MSDM model

In MSDM, the problem is approached in a more general way. In this model, a multi-structured document is defined using the following notions:

- **Document Structure (DS)**: this is a description of a document content defined to a specific use. Such structure may be, for example, a physical structure defined for a presentation goal.
- **Base Structure (BS)**: this structure is visible only internally within the multi-structured document. It is defined strictly in order to organize the content in disjoint elementary fragments. These fragments serve to reconstitute, by composition, the original content associated initially to the document structure elements.
- **Correspondence:** a correspondence is a relation between two elements of two distinct structures.

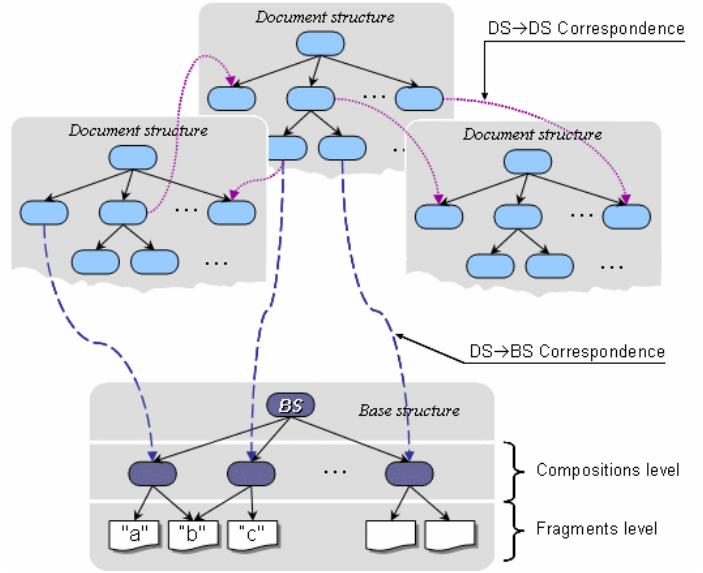

FIGURE 5. ILLUSTRATION OF THE MULTI-STRUCTURED DOCUMENT MODEL (MSDM)

As we can see in figure 5, the source of a correspondence is always an element of a document structure. If the correspondence target is an element of the base structure the correspondence is noted DS→BS. This kind of correspondence associates an element of a document structure to its content in the base structure. When the correspondence target belongs to a document structure the correspondence is noted DS→DS. The correspondences DS→DS allow to make some hided relations between document structures explicit. Such correspondence may be used to express a synonymy relation between two elements for example.

In brief we have chosen MSDM as a starting point of our work for the following reasons:

1. Its ***generality*** because it is not defined for a type of document or media and it is not limited to answer a particular use.

2. ***Borders between structures:*** In a model of representation MSDs it is not enough to melt the structures the ones in the others. It is important to be able to locate each structure easily and to clearly distinguish the elements which constitute it.

3. ***Inter-structural relations***: to facilitate and to optimize the exploitation of the document.

4. The MSDM model is ***independent of any formalism*** of encoding.

However MSDM is encoded by formalism called MultiX (Chatti 2006) that uses XML syntax. MultiX could serialize the MSDM model into XML syntax and expresses the different structures and their correspondences in a single xml file. However it still has some complexity due to its respect to XML tree model.

3.2 Why OWL

An Ontology defines a common vocabulary for researchers who need to share information in a domain. It includes machine-interpretable definitions of basic concepts in the domain and relations among them.
OWL: Web Ontology Language, is designed for use by applications that need to process the content of information instead of just presenting information to humans.
XML covers the syntactic level, but lacks support for efficient sharing of conceptualizations. The Web Ontology Language in turn supports the representation of domain knowledge using classes, properties and instances.
XML limitations in overlapping markup or complex annotations are clear as its specifications require a strict nesting of the elements. In opposite OWL could be considered as a network of vocabularies that are connected to each other via relations which corresponds to nodes and arcs in the graph concepts, as shown in figure 6.

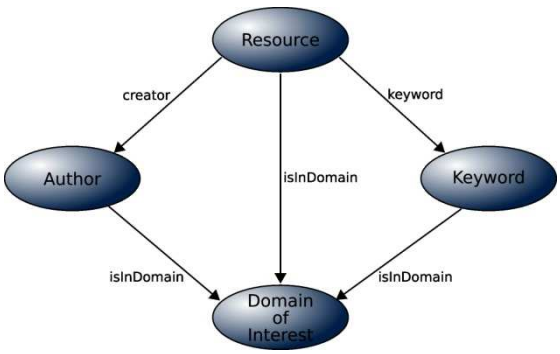

FIGURE 6. AN EXAMPLE OF OWL ONTOLOGY

Theoretically speaking, semantic web tools are suitable to fulfill all the tasks that have been traditionally done in XML.

3.3 Encoding MSDM model by means of Ontologies

We propose a textual encoding model based on the MSDM model and on tools and concepts developed within the W3C Semantic Web initiative. We illustrate a demonstrative textual encoding ontology for old manuscripts using the Ontology Web Language (OWL).

In particular we will show how the markup process can be seen as a task of knowledge representation where elements, such as words and lines, are instances of appropriate conceptual classes forming a semantic network.
To build the ontology depending on the logic of MSDM and using OWL, we consider the following points:

1. The main MSD document, Base structure (BS) and documentary structures (DS) are mapped to an OWL Class
2. All documentary structures would be considered as OWL subClass of the class DS as well as other extensions and restrictions.
3. The BS class has subclass named Fragments that is essential to specify the minimal set of disjoint content fragments which recover all superposed string segments.
4. Each DS has subclasses and relations depending on its internal structure.
5. Relations within structures are mapped to OWL ObjectProperty.
6. Further elements and attributes are mapped to an OWL DatatypeProperty or ObjectProperty, depending on their type.
7. Add OWL constraints. We could express constraints by using the quantifier restrictions:

   - The existential quantifier $\exists$, which can be read as at least one, or some. For example a multi-structured document has at least one documentary structure.
   - The universal quantifier $\forall$, which can be read as only.These constraints could be used to validate multi-structured documents.

A reasoner computes subsumption relationships between classes, and detect inconsistent classes.

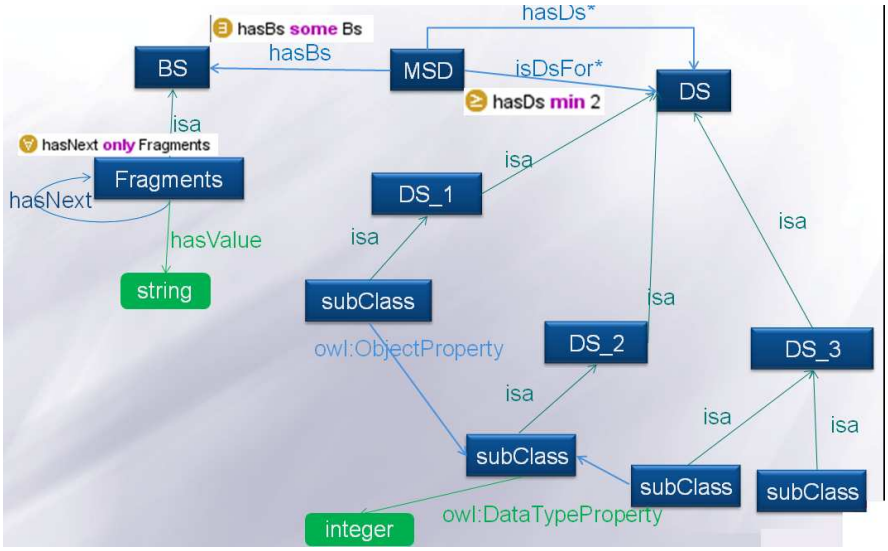

FIGURE 7. *AN APPROPRIATE ONTOLOGY FOR MSDM*

3.4 A Case study

We used Protégé-OWL (http://protege.stanford.edu/.) to build our OWL Ontologies. In addition it is necessary to have a DIG (Description Logic Implementers Group) compliant reasoner installed in order to compute subsumption relationships between classes, and detect inconsistent classes. The reasoner we used is Racer (http://www.sts.tu-harburg.de/~r.f.moeller/racer/.).
The example that we will use in this section is the old manuscript which we exploited in chapter 1.

We remind that it is an image representing an extract of an old manuscript which we associate four different structures. The first structure represents the transcription of the textual contents of the manuscript by respecting its physical structure. The second structure locates all the words which this transcription contains. The third structure marks sequences of characters of the transcription which are damaged in the original manuscript. The last structure locates rectangular zones on the image of the manuscript, which delimit particular areas containing the handwritten text.

Figure 8 illustrate the MSDM model according to our example. The first three structures share segments of textual contents. In our multi-structured document we will exploit the relations of correspondences SD→SD to locate certain segments of the textual transcription on the image of the manuscript. The first three structures will be associated, by correspondences of the type SD→SB, with the basic structure of the document to share the textual contents. Besides there are relations of correspondences of the type SD→SD established from the physical structure towards the text region structure to associate each line to its localization on the image. Between the same documentary structures, for each correspondence of localization one opposite correspondence is created to materialize the relation indicating the transcription of the text in the image. Lastly, relations of correspondences are established starting from the lexical structure towards the text region structure to locate the words cut on two lines in the image of the extract of the manuscript.

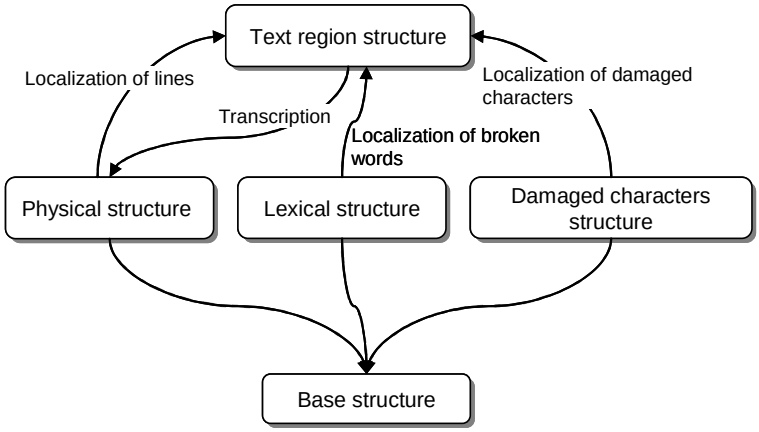

FIGURE 8. THE MULTI-STRUCTURED DOCUMENT ACCORDING TO MSDM

We built our lightweight ontology using Protégé plugin version 3.2.1.
OWL ontology consists of Individuals, Properties, and Classes. Individuals, represent objects in the domain that we are interested in, such as a certain manuscript, line number 2, second word...etc.
OWL Properties are binary relations on individuals. For example, the property hasFirstFragment might link the Line_1 to the individual Fragments_1.
Properties can have inverses. For example, the inverse of localisation is transcription. Properties can be limited to having a single value i.e. to being functional. For example an MSD (Domain) may have one and only one BS (Range). Notice that it could have more than one DS. The star symbol next to property name means “many”.
OWL classes are interpreted as sets that contain individuals. For example, Fragments, words, DS…etc. Classes are organized into a superclass-subclass hierarchy, which is also known as taxonomy. Subclasses specialize (‘are subsumed by’) their superclasses.
Figure 9 illustrates a general view of our ontology. The blue arrows correspond to owl: Object properties.
For the case, that an element in the source XML tree is always a leaf, containing only a literal and no attributes, this element in OWL language is owl:DatatypeProperty having as domain the class representing the surrounding element. Such as the description of text regions, it is always a string.
OWL constraints are added as well to the specific properties to ensure the correctness of symbol chains, e.g. that the next of a Line can only be a Line, the first symbol of a Line can only be Fragment

etc. To validate these constraints, OWL reasoners can be used directly to detect most inconsistencies. These constraints will allow validating properties on multi-structured documents.

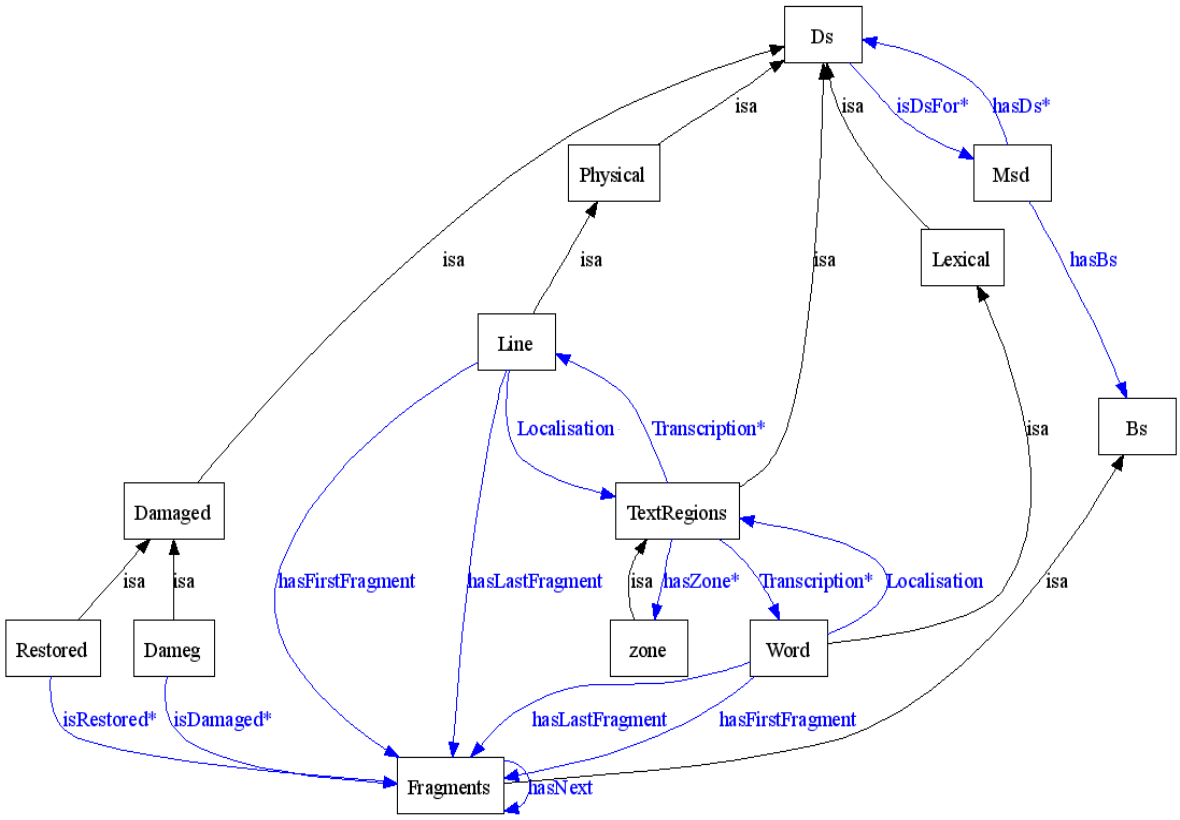

FIGURE 9. illustration of our example ontology

3.5 Discussion

Advantages of OWL formalism:

- Network of vocabularies, connected to each other via relations that correspond to nodes and arcs in the graph concepts.
- Conceptualization: representation of domain knowledge using classes, properties and instances.
- Enable reuse of domain knowledge.
- Regulate and validate the interconnection between resources.
- Possibility to query multi-structured documents with SPARQL or QL

4 CONCLUSION AND PERSPECTIVES

In this work we studied the problem of overlapping in Multi-structured documents (MSDs).
Multi-structured documents are documents whose structure is composed of a set of concurrent hierarchical structures.
We found that the methodology proposed to encode textual documents by means of graph is quite interesting that open a new horizon in the field of multi-structured documents.
We chose MSDM (Multi-structured Document Model) as a generic model and tried to encode it by Ontology web language (OWL), by building ontology classes that depend on the logical model of MSDM.

Future works:

- Enhance the proposed ontology and add more needed aspects like semantic structure, documents containing text and images ...
- Querying the proposed model by SPARQL, by defining custom operators which can be then invoked within a query using the filter construct.
    - Example of multi-structured query: find all words cut by end of line…

References

Battle, S. (2004). Round-tripping between XML and RDF. International Semantic Web Conference(ISWC). Hiroshima, Japan.

Bruno, E. and Murisasco, E. (2006) MSXD : Représenter et interroger des documents XML textuels multistructurés.. Journées de travail interdisciplinaire autour des documents multi structurés, Giens.

Chatti N. (2006). Documents multi-structurés: De la modélisation vers l'exploitation. PhD Thesis. INSA de Lyon, France.

Chatti, N. Kaouk, S and Calabretto, S. (2007). MultiX: an XML-based formalism to encode multi-structured documents. Extreme Markup Language Conference, Montreal-Canada.

Ferdinand, M. Zirpins, C. and Trastour D. (2004). Lifting XML Schema to OWL. 4th International Conference on Web Engineering, ICWE, Springer Heidelberg, Munich, Germany.

Le Maitre, J. (2006). Describing Multi-structured XML Documents by Means of Delay Nodes. In Proceedings of the 2006 ACM Symposium on Document Engineering (DocEng 2006), pp. 155-164, Amsterdam, the Netherlands.

McQueen, S. and Huitfeldt, C. (2000) GODDAG: A Data Structure for Overlapping Hierarchies. In Proceeding of the 5th International Workshop on the Principles of Digital Document Processing (PODDP 2000). Lecture Notes in Computer Science 2023. Springer Verlag, 139-16.

Tenison, J. and Piez,W. (2002) Layered markup and annotation language (lmnl). In The Late breaking paper presented at Extreme Markup.

Tummarello, G. Morbidoni, C. and Pierazzo, E. (2005). Toward Textual Encoding Based on RDF. In Proceedings of ELPUB2005, Leuven-Heverlee(Belgium).

Tummarello, G. Morbidoni, C. Kepler, F. Piazza, F. and Puliti, P. (2007) A proposal for Textual Encoding based on Semantic Web tools. International Conference on Semantic web and Digital libraries (ICSD 2007). Bangalore.